Vol.2018-IOT-43 No.13
Vol.2018-SPT-30 No.13
2018/9/28

# 機械学習を用いた巧妙なフィッシングメールの識別方法の検討

髙橋昌士　猪俣敦夫
東京電機大学

**概要**：フィッシングメールによる機密情報の搾取は非常に多くの人にとって驚異である.現在の主な対策は自然言語処理フィルタやファイルのハッシュ情報を活用したスパムフィルタリングを行っているが,検知精度はフィルタリングを提供する側の知見に依存する.そこで今回は巧妙なフィッシングメールをメールのヘッダ情報を特徴にして,機械学習のロジスティック回帰のアルゴリズムで識別することを検討した.取り組みの結果としては 3031 通のサンプルに対し約 70%の確立で識別することができた.

**キーワード**：情報セキュリティ、フィッシングメール、機械学習

# Consideration of a clever phishing mail identification method using machine learning

Tokyo Denki University　MASASHI TAKAHASHI　ATSUO INOMATA

***Abstract***: The exploitation of confidential information by phishing e-mail is threat for a large number of people. The current main method is spam filtering utilizing natural language processing filter and file hash information, but detection accuracy depends on knowledge of the provider. So we consider classification method phishing e-mails using mail header information by logistic regression algorithm of machine learning. As a result, we were able to identify 3031 samples by about 70% establishment.

***Keywords***: information security, phishing mail, machine learning

## 1. はじめに

現在 SNS を中心としたコミュニケーションツールが利用の割合を徐々に伸ばしているが、電子メールはコミュニケーションツールの中で依然として多くの利用者がおり、被害件数も多い[1].その電子メールを入り口としたサイバー攻撃手法は多岐に渡り、代表的な手法としてスパムメールによる DoS 攻撃、フィッシングメールによる機密情報の搾取、標的型攻撃による LAN ネットワークへの侵入とマルウェアを利用した乗っ取り、ビジネスメール詐欺による金銭要求などがある.そのため、多くのシステム管理者は電子メールを利用者が安全に利用できるように送信側の対策と受信側の対策を運用している[2].受信側の対策としてはブロッキングやフィルタリング技術の利用を前提にシステム運用をしており、その手法も攻撃手法同様に多岐に渡る.フィルタリングによる対策に関しては代表的な手法としてスパムメールを本文の記述内容から自然言語処理の活用によって判定したり、添付されているファイルのハッシュから悪性ファイルを検知するアンチウイルス技術の活用、実行した添付ファイルの振る舞いから悪性ファイルを検知するサンドボックス技術などがある[2][3].これらを運用し、電子メールを利用したサイバー攻撃に対応するときの課題として、巧妙なフィッシングメールは実際のサービス事業者から送付されてくる広告メールや通知メールと見た目からは区別がつきにくいことから、本当の広告や通知メールだと勘違いしやすい点や添付ファイルがついてない場合も多く、URL のリンクに注意しなければいけない点も巧妙なフィッシングメールの特徴である.これらのメールを既存の技術でフィルタリングしようとするとなかなか難しい[4].そこで今回は巧妙なフィッシングメールのヘッダ部分に着目し、受信したメールのヘッダ情報を特徴として識別する機械学習モデルを検討した.

## 2. 巧妙なフィッシングメールの定義

まずフィッシングメールを含むメールの全体的な分類と危険度を定義する

電子メール全体に対する分類は 4 種類に分けられる[2].

| | |
|---|---|
| 1 | 送信者と受信者で情報交換を行う利用者メール |
| 2 | 事業者等が利用者へ広告や通知を行う広告メール |
| 3 | 攻撃者が機密情報搾取や金銭要求を狙う詐欺メール |
| 4 | 攻撃者が不正侵入等を狙うマルウェア添付メール |

また、これらのメールについて危険度を定義する[3].

| | |
|---|---|
| 小 | 利用者メール、広告メール |
| 中 | 詐欺メール |
| 大 | マルウェア添付メール |

この定義からフィッシングメールはメールの分類が 3 に分類され、危険度が中に値するものとする.しかし、フィッシ

Vol.2018-IOT-43 No.13
Vol.2018-SPT-30 No.13
2018/9/28

ングメールの内容が巧妙かどうかについては区別が無いため、ここではメール受信者が受信したメール本文の「見た目」から広告メールと見分けができない、またはしにくいものを巧妙なフィッシングメールとする.

## 3. 関連研究

3-1.既存のフィルタリング技術について

これらフィッシングメールを含むスパムメールのフィルタリングに関する研究は昔から数多く行われており、実用化されているものも多数ある.代表的なものを 2 つ紹介する.

1 つ目は自然言語処理によるスパムフィルタ[3]がある.自然言語処理は形態素解析などの技術を用い、メールの文面を単語ベースに解析し、文面上に出現する単語のスパムメールである可能性を計算する.最終的に単語ごとのスパムである可能性を合算し、メール全体の「スパムらしさの確率」を元に振り分けを行う.形態素解析には MeCab[5]など専用の解析モジュールを用い、単語の抽出を行う.また合算の計算にはベイズ定理などがよく用いられる.

2 つ目はファイルハッシュによるフィルタリングがある[6].これは受信したメールに添付されているファイルについて、予め用意したパターンファイルと照合し、合致した場合は不審なファイルとして判定することでそのファイルが添付されたメールはスパムメールであると判定する.

これらの手法を利用し、巧妙なフィッシングメールを検出しようとすると、いずれも課題がある.確率的な処理は本文に利用されている文字から特徴を見出すため、本文中の内容に特徴がなければ判別が難しい.また、ファイルハッシュによる検知もファイルが添付されておらず、URL だけになると検知が難しいという課題を持つ[7].

現在、電子メールに関するセキュリティサービスを提供する事業者の多くは、これらの手法に加えてサービス提供事業者独自の知見を組み合わせた総合的な電子メールセキュリティサービスとして利用者へ提供していることが多い.

3-2.関連研究について

最近は機械学習を用いてフィッシングメールを検出する研究が多い[8].また、11 の静的なルールでフィッシングメールを判定するという研究もある[9].11 の静的なルールのうち、1 つに「Received」というヘッダをチェックし、ヘッダパスに FROM やメッセージ本文と同じドメインのサーバまたはメールユーザーエージェントが含まれていない場合はフィッシングメールである要素であると判定している.これらの研究からもフィッシングメールの判定に機械学習を用いることやヘッダ情報を用いることの有効性が示されている.

## 4. 特徴量の抽出と学習アルゴリズムの選択

4-1.特徴量について

ここまで電子メールを用いた攻撃の脅威の現状や関連研究について述べてきた.ここでは受信したメールのヘッダから機械学習で扱う特徴量をどのように抽出したかを示す.今回のサンプルとして収集した 3031 件のメールにスタンプされていたヘッダは全部で 503 種類あった. (図 1 参照).これらのヘッダから特徴量を抽出する手順を以下に示す.

1. サンプルをフィッシングメールか否かに分類
2. グループごとに 50%以上スタンプされているヘッダを抽出.
3. 抽出ヘッダから受信環境に依存するヘッダと受信環境には依存しないヘッダに分類(図 2 参照).
4. 受信環境に依存するヘッダを除外
5. 残ったヘッダの値をデータ長へ計算した結果が特徴量.

ヘッダの値から、さらに特徴を見出すためには DNS や WHOIS といった他のシステムの登録情報を活用することも考えられたが、今回はできるだけ受信者の環境だけで検討するためにヘッダの値による判定ではなく、ヘッダの値ごとにデータ長を算出し、特徴量とした(図 3 参照).

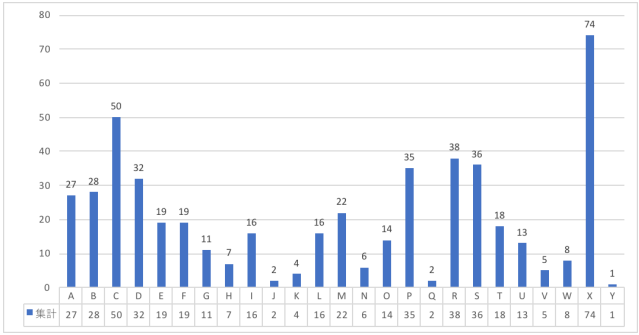



図 1 メールヘッダの頭文字ごとの統計

| メール内に該当ヘッダが含まれる割合 | 通常メール | フィッシングメール | 受信時の環境に依存 |
|---|---|---|---|
| RECEIVED | 100.00% | 100.00% | しない |
| FROM | 100.00% | 100.00% | しない |
| DATE | 99.96% | 100.00% | しない |
| CONTENT-TYPE | 99.96% | 100.00% | しない |
| SUBJECT | 99.92% | 100.00% | しない |
| TO | 99.37% | 100.00% | しない |
| RETURN-PATH | 99.09% | 99.00% | しない |
| MESSAGE-ID | 97.36% | 100.00% | しない |
| MIME-VERSION | 82.28% | 100.00% | しない |
| X-ANTIVIRUS | 76.60% | 66.93% | する |
| X-ANTIVIRUS-STATUS | 76.60% | 66.93% | する |
| DELIVERED-TO | 73.64% | 7.62% | しない |
| X-ORIGINAL-TO | 72.30% | 6.81% | しない |
| CONTENT-TRANSFER-ENCODING | 65.51% | 35.87% | しない |
| DKIM-SIGNATURE | 55.21% | 0.80% | しない |
| REPLY-TO | 52.05% | 0.60% | しない |
| X-MAILER | 24.63% | 73.95% | しない |
| X-MOZILLA-STATUS | 9.55% | 66.93% | する |
| X-MOZILLA-STATUS2 | 9.59% | 66.93% | する |
| THREAD-INDEX | 5.96% | 64.33% | する |
| X-OLKEID | 5.17% | 63.73% | する |
| X-SPAMINFO | 5.09% | 63.73% | する |
| X-ASE-REPORT | 4.38% | 63.13% | する |
| X-MS-TNEF-CORRELATOR | 3.40% | 62.32% | する |

図 2 各メール内の特徴的なヘッダの含有率

| ヘッダ名 | ヘッダの意味 |
|---|---|
| RETURN-PATH | メールが届かなかったときのエラー通知先アドレス |
| RECEIVED | メールを受信するまでに通過してきたメールサーバ情報 |
| FROM | 送信元メールアドレス |
| DATE | 受信日時 |
| SUBJECT | メールのタイトル |
| TO | 送信先メールアドレス |
| MESSAGE-ID | 受信したメールのID |
| CONTENT-TYPE | メール本文の構造や漢字コードの指定を行う |
| REPLY-TO | メールに返信するときの返信先アドレス |
| CONTENT-TRANSFER-ENCODING | MIMEのあるメールの表現方法を指定 |
| X-MAILER | 送信元メーラの種類 |
| DKIM-SIGNATURE | 送信元サーバ認証に利用するヘッダ |
| X-ORIGINAL-TO | 送信者が本来送信した宛先 |
| DELIVERED-TO | 送信者が本来送信した宛先から別のアドレスに転送された宛先 |
| MINE-□VERSION | MIMEのバージョン番号を示す。現在は1.0しか存在しない。 |

図 3　特徴量としたメールヘッダとヘッダの解説[2]

4-2.機械学習のアルゴリズムの選択について

機械学習を用いてモデルを構築する手法には主に教師あり学習、教師なし学習、強化学習がある.一般的には機械学習を行う目的により利用する手法が異なる. サンプルを予め定義したクラスへ分類することや数値的な未来予測を行うような回帰を行うことが目的の場合は教師あり学習を採用する場合が多い.分類するクラスが不明確なサンプルをいくつかのクラスに分けるクラスタリングを行う場合は教師なし学習を採用する場合が多く、答えは明確だが答えまでのプロセスが不明確で、プロセスを最適化したい場合は強化学習を採用することが多い[10].

今回は巧妙なフィッシングメールを機械学習により識別できるモデルを検討することが目的であるため「教師あり学習」による２値分類を行うこととした.識別に利用する教師データは予め著者によってサンプルをフィッシングメールであるか否か分類し、ラベル付けしたものを利用する.アルゴリズムはフィッシングメールであるか否かを分類する 2 値分類である点から最も一般的なロジスティック回帰を選択した.

## 5. 実験

5-1.実験の環境について

本実験は以下の環境を用いて行った.(図 4 参照)

| 環境 | 用途 |
|---|---|
| MacBookPro(2016) | PC |
| Python(anaconda) | プログラム言語,versionは3.6.2 |
| pycharm professional/jupyter notebook | 開発環境 |
| codecs | python用モジュール,ファイルの読み込み、出力 |
| re | pythonモジュール:正規表現の利用 |
| subprocess | pythonモジュール:osコマンドの利用 |
| email.parser.Parser | pythonモジュール:emailの解析モジュール |
| email.header. decode_header | pythonモジュール:emailのヘッダ情報のデコード |
| pandas | pythonモジュール:行列計算用モジュール |
| seaborn | pythonモジュール:多様なグラフ作成モジュール |
| matplotlib.pyplot | pythonモジュール:グラフ作成エンジンモジュール |
| sklearn.model_selection. train_test_split | pythonモジュール:機械学習用(訓練データとテストデータの分類) |
| sklearn.linear_model. LogisticRegression | pythonモジュール:機械学習用(ロジスティック回帰モデルの作成) |
| sklearn.metrics. classification_report | pythonモジュール:機械学習用(分類結果の確認用1) |
| sklearn.metrics.confusion_matrix | pythonモジュール:機械学習用(分類結果の確認用2) |

図 4　利用した実験環境

5-2.前処理について

機械学習にデータを読み込ませるために以下の前処理 1 から 10 を行った.

1. 収集した 3031 通のメールを予め作成したフォルダ(今回の実験では mail フォルダおよび malicious_mail フォルダとした)へ巧妙なフィッシングメール以外と巧妙なフィッシングメールに振り分けながら格納.この際振り分けは定義に基づいた振り分けを手作業で著者が実施(図 5 参照).

| 振り分けカテゴリ | 件数 | 格納フォルダ |
|---|---|---|
| フィッシングメール以外 | 2533 | mail |
| フィッシングメール | 498 | malicious_mail |

図 5　振り分け結果

2. 各フォルダに格納したメールを全て読み込み、サンプルメールの一覧リストを作成.
3. 作成したリストのメールを email.parser.Parser モジュールで 1 通ずつ全て読み込み、ユニークなヘッダリストを作成.
4. ユニークなヘッダリストと、メールから読み込んだヘッダを突合し、合致したところでヘッダの値をファイルへ記入.それ以外は null をファイルへ記入.
5. ヘッダの値が base64 でエンコードされている場合は email.header.decode_header モジュールで読み込み、元の文字コードデータへ戻し、ファイルへ記入.
6. 作成するデータの列に malicious という列を作成し、mail フォルダから読み込んだ場合は 0,malicious_mail フォルダから読み込んだ場合は 1 を記入.
7. 各データを記入したファイルは,(カンマ)区切りで作成し、CSV ファイルとして保存.
8. 工程 1 から 8 で作成した CSV ファイルを読み込み、全てのデータに対し、データ長を計算しファイルへ記入.この際 null データは 0 に置換.
9. 4-1 で定義したヘッダ情報をのぞき、列を削除.
10. 作成した CSV ファイルを pandas モジュールで読み込む.(図 6 参照)(図 7 参照)

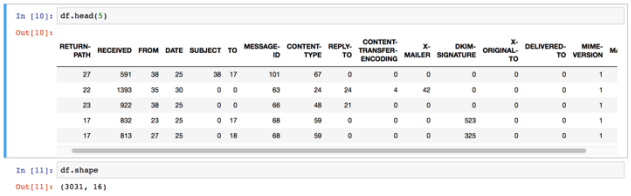
```
In [8]: df = pd.read_csv("read_data.csv")

In [9]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3031 entries, 0 to 3030
Data columns (total 16 columns):
RETURN-PATH                 3031 non-null int64
RECEIVED                    3031 non-null int64
FROM                        3031 non-null int64
DATE                        3031 non-null int64
SUBJECT                     3031 non-null int64
TO                          3031 non-null int64
MESSAGE-ID                  3031 non-null int64
CONTENT-TYPE                3031 non-null int64
REPLY-TO                    3031 non-null int64
CONTENT-TRANSFER-ENCODING   3031 non-null int64
X-MAILER                    3031 non-null int64
DKIM-SIGNATURE              3031 non-null int64
X-ORIGINAL-TO               3031 non-null int64
DELIVERED-TO                3031 non-null int64
MIME-VERSION                3031 non-null int64
MALICIOUS                   3031 non-null int64
dtypes: int64(16)
memory usage: 379.0 KB

In [11]: df['MALICIOUS'].value_counts()
Out[11]: 0    2533
         1     498
         Name: MALICIOUS, dtype: int64
```

図 6　pandas モジュールで読み込んだヘッダ

Vol.2018-IOT-43 No.13
Vol.2018-SPT-30 No.13
2018/9/28



```
In [10]: df.head(5)
Out[10]:
```

| RETURN-PATH | RECEIVED | FROM | DATE | SUBJECT | TO | MESSAGE-ID | CONTENT-TYPE | REPLY-TO | CONTENT-TRANSFER-ENCODING | X-MAILER | DKIM-SIGNATURE | X-ORIGINAL-TO | DELIVERED-TO | MIME-VERSION | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 27 | 591 | 38 | 25 | 38 | 17 | 101 | 67 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 22 | 1393 | 35 | 30 | 0 | 0 | 63 | 24 | 24 | 4 | 42 | 0 | 0 | 0 | 1 | |
| 23 | 922 | 38 | 25 | 0 | 0 | 66 | 48 | 21 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 17 | 832 | 23 | 25 | 0 | 17 | 68 | 59 | 0 | 0 | 0 | 523 | 0 | 0 | 1 | |
| 17 | 813 | 27 | 25 | 0 | 18 | 68 | 59 | 0 | 0 | 0 | 325 | 0 | 0 | 1 | |

```
In [11]: df.shape
Out[11]: (3031, 16)
```

図 7　pandas モジュールで読み込んだヘッダ値

5-3.実験の進め方について

本実験は以下の 1 から 7 の進め方で実施する.

1. 5-2 で読み込んだデータについて、まずは各データの分布、データ間の関係性を目視で確認するため、seaborn モジュールの pairplot グラフで可視化.この pairplot グラフでは全要素を総当たりで散布図表記させる散布図行列を作成ことができ、比較的容易に要素および要素間の関係性を分析することが可能.
2. 可視化されたグラフの中で特に偏りや相関を分析.(図 8 参照)
3. 散布図の分析から得られた情報を踏まえ、実験方針を作成(図 9 参照).
4. 方針に従い、モデルを構築するために必要な列のみ残し、残りを削除.
5. 全体の 70%を機械学習の訓練データ、残り 30%をテストデータに分類.
6. 作成した訓練データでロジスティック回帰モデルを構築.
7. 構築したモデルにテストデータを読み込ませ、予め付与されているラベルと整合性を評価.

```
In [17]: sns.pairplot(data=df,vars=['DKIM-SIGNATURE','X-ORIGINAL-TO', 'REPLY-TO'],hue='MALICIOUS')
Out[17]: <seaborn.axisgrid.PairGrid at 0x11140f978>
```

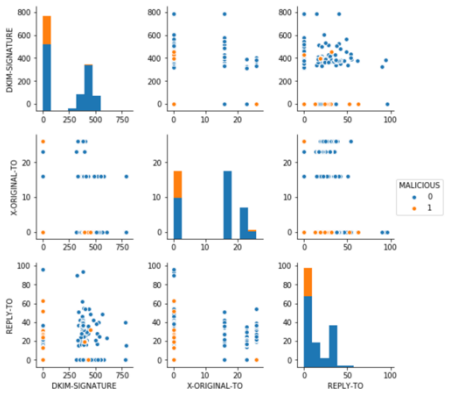



図 8　特に偏りが見られたヘッダデータの散布図行列

| 実験項番 | 実験するモデルの考え方 | 該当するヘッダ数 |
|---|---|---|
| 1 | 抽出したヘッダを全て利用したモデル | 15 |
| 2 | フィッシングメール以外とフィッシングメールのどちらにも80%以上の含有率が確認できたヘッダデータを利用したモデル | 9 |
| 3 | フィッシングメール以外とフィッシングメールで含有率の差分が50%未満のヘッダデータを利用したモデル | 11 |
| 4 | 実験3で削除したヘッダのみを利用したモデル | 6 |

図 9　実験方針

5-3.実験結果について

5-2 で示した実験の進め方で実験を進めた結果を示す.

5-3-1.実験方針の項番 1 に従い、実験した結果 (図 10 参照)

```
In [27]: print(classification_report(y_test,predixtions))
         # 79%で振り分け成功
                      precision    recall  f1-score   support

                   0       0.99      0.95      0.97       760
                   1       0.79      0.95      0.86       150

         avg / total       0.96      0.95      0.95       910

In [28]: confusion_matrix(y_test,predixtions)
Out[28]: array([[722,  38],
                [  7, 143]])
```

図 10　抽出したヘッダを全て利用したモデルの実験結果

5-3-1 の実験は巧妙なフィッシングメールを 79%の確率で正しく識別することができた.

5-3-2. 実験方針の項番 2 に従い、実験した結果.(図 11 参照)

```
In [34]: X.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3031 entries, 0 to 3030
Data columns (total 9 columns):
RETURN-PATH     3031 non-null int64
RECEIVED        3031 non-null int64
FROM            3031 non-null int64
DATE            3031 non-null int64
SUBJECT         3031 non-null int64
TO              3031 non-null int64
MESSAGE-ID      3031 non-null int64
CONTENT-TYPE    3031 non-null int64
MIME-VERSION    3031 non-null int64
dtypes: int64(9)
memory usage: 213.2 KB

In [38]: print(classification_report(y_test,predixtions))
         #振り分け成功確率が57%に減少
                      precision    recall  f1-score   support

                   0       0.92      0.91      0.92       760
                   1       0.57      0.58      0.58       150

         avg / total       0.86      0.86      0.86       910

In [39]: confusion_matrix(y_test,predixtions)
Out[39]: array([[695,  65],
                [ 63,  87]])
```

図 11　80%以上の含有率のヘッダのみの実験結果

5-3-2 の実験は巧妙なフィッシングメールを 57%の確率で正しく識別することができた.5-3-1 の実験結果に比べると識別の成功率が 22%減少している.

5-3-3. 実験方針の項番 3 に従い、実験した結果 (図 12 参照)

```
In [20]: X.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3031 entries, 0 to 3030
Data columns (total 11 columns):
RETURN-PATH                  3031 non-null int64
RECEIVED                     3031 non-null int64
FROM                         3031 non-null int64
DATE                         3031 non-null int64
SUBJECT                      3031 non-null int64
TO                           3031 non-null int64
MESSAGE-ID                   3031 non-null int64
CONTENT-TYPE                 3031 non-null int64
CONTENT-TRANSFER-ENCODING    3031 non-null int64
X-MAILER                     3031 non-null int64
MIME-VERSION                 3031 non-null int64
dtypes: int64(11)
memory usage: 260.6 KB

In [21]: print(classification_report(y_test,predixtions))
         #振り分け成功確率は58%に減少
             precision    recall  f1-score   support

          0       0.92      0.92      0.92       760
          1       0.58      0.58      0.58       150

avg / total       0.86      0.86      0.86       910

In [22]: confusion_matrix(y_test,predixtions)
Out[22]: array([[696,  64],
                [ 63,  87]])
```

図 12　含有率差が 50%以上のヘッダを削除した実験結果

5-3-3 の実験は巧妙なフィッシングメールを 58%の確率で正しく識別することができた. 5-3-1 の実験結果に比べると成功率が 21%減少している.

5-3-4. 実験方針の項番 4 に従い、実験した結果 (図 13 参照)

```
In [52]: X.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3031 entries, 0 to 3030
Data columns (total 6 columns):
REPLY-TO                     3031 non-null int64
CONTENT-TRANSFER-ENCODING    3031 non-null int64
X-MAILER                     3031 non-null int64
DKIM-SIGNATURE               3031 non-null int64
X-ORIGINAL-TO                3031 non-null int64
DELIVERED-TO                 3031 non-null int64
dtypes: int64(6)
memory usage: 142.2 KB

In [56]: print(classification_report(y_test,predixtions))
         #振り分け成功確率は70%に減少
             precision    recall  f1-score   support

          0       1.00      0.92      0.96       760
          1       0.70      0.99      0.82       150

avg / total       0.95      0.93      0.93       910

In [57]: confusion_matrix(y_test,predixtions)
Out[57]: array([[698,  62],
                [  2, 148]])
```

図 13　含有率差が 50%以上のヘッダを残した実験結果

5-3-4 の実験は巧妙なフィッシングメールを 70%の確率で正しく識別することができた. 5-3-1 の実験結果に比べると成功率が 8%減少している.

5-3-5.モデルの評価について

今回の実験では以下の観点でモデルを評価(図 14 参照)

| 評価項目名 | Precision | Recall | F1-score |
|---|---|---|---|
| 項目の解説 | ポジティブなサンプルがTNやFNに分類された割合 | ポジティブなサンプルがFNやFPに分類されなかった割合 | 精度（Precision）と検出率（Recall）をバランス良く持ち合わせているかを示す指標◌1がバランスが良い. |
| 評価の見方 | 0.00～1.00までで算出され、1.00はテストデータが全てTPに正しく分類されたことを示す | 0.00～1.00までで算出され、1.00はテストデータが全てTPに正しく分類されたことを示す | 0.00～1.00の間で計算され、1.00に近いほどモデルのばrンスがよいことを示す |
| 計算式 | TP/(TP+FP) | TP/(TP＋FN) | (2×precision×recall)/(precision+recall) |

TruePositiveはTPと表す
TrueNegativeはTNと表す
FalsePositiveはFPと表す
FalseNegativeはFNと表す

図 14　評価項目と見方、計算式の一覧[10]

## 6. 考察

5-3-1 から 5-3-4 の実験を通じて得られた結果を考察した.

6-1.実験結果について

5−3−1 は precious が 79%、recall が 95%と今回の実験の中においては precious の値が一番高いモデルとなった.要因として、巧妙なフィッシングメールとそれ以外のヘッダではヘッダ内で偏りが見られる部分が多く、それぞれの偏りを特徴として積み重ねた結果、比較的高い識別率を持つモデルになったと想定される.また、recall の値が 95%であることも同様の理由と考えられる.

5-3-2 は precious が 57%、recall が 58%と今回の実験の中では最も低い実験結果だった.これは 5-3-1 で考察したような比較的偏りが見られたヘッダが取り除かれたことによる結果と考える.つまり、巧妙なフィッシングメールとそれ以外を識別するためにはどちらのメールにも含有されるヘッダのヘッダ長情報だけでは識別するのが困難であることを示していると考える.

5-3-3 は precious および recall が 58%と 5-3-2 とほぼ同様の結果となった.これは 5-3-2 と比べて差分となったヘッダ(CONTENT-TRANSFER-ENCODING、X-MAILER)もこの場合においては有効な識別要素にならなかったことを示していると考える.

5-3-4 は precious が 70%と 5-3-1 と比較すると 8%ほど減少しているが recall は 4%上昇し、99%となった.モデル構築に利用しているヘッダは 6 に減少しているが、比較的高い識別機能を有していると考えられる.特に recall が上昇したことは Positive 群の判別に対して、含有率が低いヘッダを利用することの有効性を示していると考える.つまり、巧妙なフィッシングメールを識別するモデルを構築するにあたり、含有率が低いヘッダを導くことが positive 群を正しく識別する要素になり得ると考える.

6-2.サンプルの妥当性について

サンプルの妥当性を考察するにあたり、2 つの観点から考察する.

1 つはサンプルデータ数、バランスの妥当性である.今回はサンプルを 3031 体収集し実験を行ったが、少々バランスがよくなかったと考える.巧妙なフィッシングメールのサンプルは 498 体であり全体の 1/6 程度であった.全体の 1/2 程度の 1500 体ほど収集し実験できれば理想的だった.このことは実験結果にも現れており、各実験結果の avg/total の値が高くでていることはフィッシングメール以外の分類が成功しているためであると考えられ、サンプルのバランスがよくないことがモデル全体の評価値に影響していると考える.

もう 1 つはサンプルのメールデータの網羅性だが、今回利用したサンプルは主に国内向けに送信されていることが観測されているものを多く採用した .主な内容に大手コンピュータ OS メーカ、オンラインショッピング,大手銀行、配送業者、SNS(図 15 参照)と多種に渡っており、ターゲットにした巧妙なフィッシングメールのサンプルの網羅性という点では満たしていると考える.

| タイトル | 個数 |
|---|---|
| [Spam]あなたの[redacted]のセキュリティ質問を再設定してください。 | 84 |
| [Spam]カード利用のお知らせ | 80 |
| [Spam]【重要】カスタマセンターからのご案内【[redacted]株式会社】 | 20 |
| [redacted]IDアカウントを回復してください | 16 |
| [Spam][[redacted]注文内容ご確認（自動配信メール） | 14 |
| [Spam]口座振替日のご案内[[redacted]株式会社][redacted] | 14 |
| [Spam][[redacted]ご請求予定金額のご案内 | 10 |
| [Spam][redacted]信託銀行 - 口座開設申込受付 | 8 |
| [Spam]M[redacted]アカウントの不審なサインイン | 8 |
| [redacted]注文内容ご確認（自動配信メール） | 8 |
| [Spam][[redacted]]会員情報変更のお知らせ | 8 |
| Spam [redacted]注文内容ご確認（自動配信メール） | 7 |
| [Spam]【重要】定期的なID・パスワード変更のお願い／コンピュータウイルスにご注意を | 6 |
| Youritemwaspurchased | 5 |
| Notpossibletomakedelivery | 5 |
| [Spam][redacted]ご購入手続き完了のお | 4 |
| Deliveryproblemsnotification | 4 |
| Criticalsecurityalertforyourlinkedaccount | 4 |
| Directmessage | 4 |
| [Spam]【[redacted]】商品発送のお知らせ | 4 |
| [Spam]【[redacted]銀行】リザーブドプランカードお申込み | 4 |
| Criticalalertforyourlinkedaccount | 4 |
| [Spam]Your$13382.48Deposit | 4 |
| Criticalsecurityalertforyourprofile | 4 |
| [Spam][[redacted]]:請求内容確定のご案内 | 4 |
| [Spam]「[redacted]カードローン」仮申し込みの審査結果のご連絡 | 4 |
| [Spam]お支払いが確認できませんでした | 4 |
| [Spam]クレジットカード決済が完了しました | 4 |
| [Spam]ご注文ありがとうございました | 4 |

図 15　実験に利用した巧妙なフィッシングメールの subject ごとのメール数

6-3.他フィルタリング技術との比較、優位性の有無

3.関連研究で述べた他技術と比較として言えることは本実験で示した手法は受信者の環境だけを利用し、フィルタリングサービス提供者の専門的な知見を利用することなく巧妙なフィッシングメールを識別できることが利点である.しかし、識別の成功率はまだ低いことから、モデル構築に利用する特徴量の抽出の方法や利用するヘッダ等について研究、改善の余地があると考える.

6-4.本分類モデルについて

今回は 2 値分類のモデル構築ということで最も一般的なロジスティック回帰をアルゴリズムに採用し検討を行った.そのため、その他のアルゴリズムを利用した場合については触れていない.ロジスティック回帰は結果の判定にシグモイド関数、パラメータの更新式には尤度関数を用いる[10].つまり、データ列ごとに尤度の値を求め、合算した合計値をシグモイド関数の結果によりフィッシングメールか否かへ振り分けている.今回の 5-3-2 で行った実験で試したように識別するデータ群に特徴差が少ないデータを用いてモデルを構築する場合はアルゴリズムの仕組みが影響し、あまりよい精度がでなかった.この点からもその他のアルゴリズムを用いた場合の実験には実験余地があるため、その他のアルゴリズムを用いた場合に結果はどう変わるのかという点と今回の目的に最も適したアルゴリズムを選定する研究を継続することが必要であると考える.

## 7. おわりに

本稿では既存のフィルタング技術では検知しづらい巧妙なフィッシングメールへの脅威対策として、受信したメールのヘッダ情報の値をデータ長に変換したデータを特徴量とし、ロジスティック回帰のアルゴリズムを用いて識別する検討を行った.今回の実験では高い識別率を実現することはできなかったが、新たな可能性としては今後も検討が続けられそうな結果を示すことができた.今後はさらなる精度向上に向け、サンプル収集の強化、モデル構築に利用するヘッダ情報のさらなる精査、他学習アルゴリズムによる実験を継続していきたい.

## 参考文献

[1] “総務省　平成 30 年版　情報通信白書”
[2] “迷惑メール対策推進協議会　迷惑メール対策ハンドブック 2017”
[3] “スパムメールの教科書” 渡部綾太(著) 愛甲健二(著)
[4] “総務省　国民のための情報セキュリティサイト”http://www.soumu.go.jp/main_sosiki/joho_tsusin/security/enduser/security01/05.html
[5] MeCab 公式サイト http://taku910.github.io/mecab/
[6] iij ウイルス検出・駆除サービス https://www.iij.ad.jp/biz/po/vp.html
[7] “スパム対策の取り組みにおける課題” http://salt.iajapan.org/wpmu/anti_spam/admin/reference/report_01/ 原文: Dave Crocker（Brandenburg InternetWorking 社）著 Challenges in Anti-Spam Efforts （The Internet Protocol Journal – Volume 8, Number 4）
[8] “Detection of online phishing email using dynamic evolving neural network based on reinforcement learning”. Sami Smadi, Nauman Aslam ,Li Zhang
[9] “Phishwish a simple and stateless phishing filter” Debra L. Cook, Vijay K. Gurbani, and Michael Daniluk
[10] “Python 機械学習プログラミング 達人データサイエンティストによる理論と実践 (impress top gear)” Sebastian Raschka (著),株式会社クイープ (翻訳), 福島真太朗 (翻訳)